IE 410 – INTRODUCTION TO ROBOTICS

Lab-5 report

Running talker and listener codes

Team M410:

201901011 – HIMANSHU DUDHATRA
201901024 – DHAVALSINH RAJ
201901100 – SHUBHAM PATEL
201901145 – GARGEY PATEL

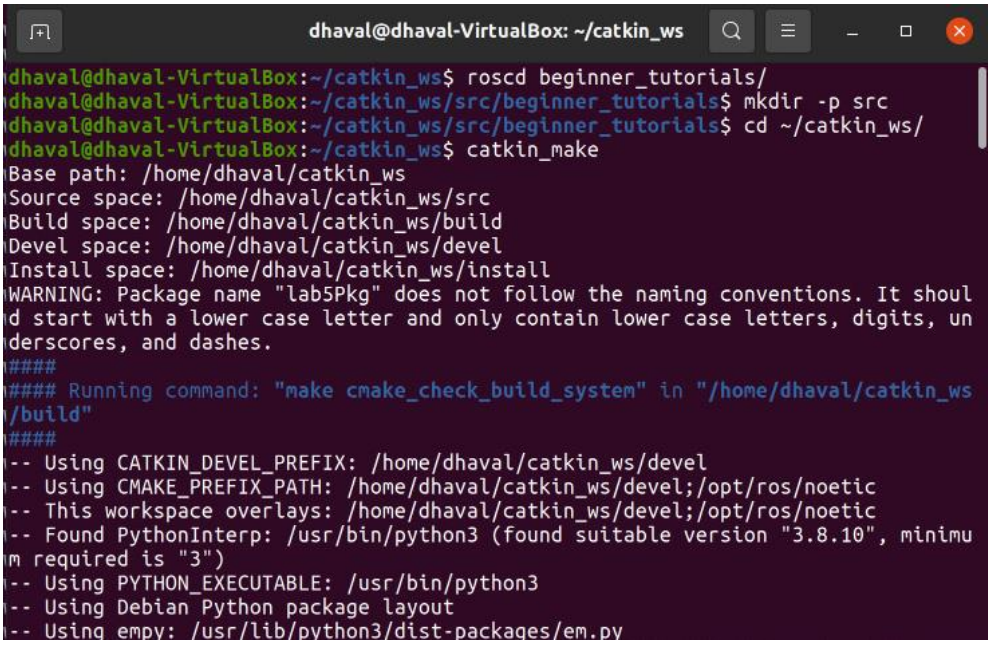
CPP

- **Change directories to your beginner_tutorials package you created in your catkin workspace previous tutorials:**

```
roscd beginner_tutorials
```

- **Create a `src` directory in the beginner_tutorials package directory:**

```
mkdir -p src
```



```
dhaval@dhaval-VirtualBox:~/catkin_ws$ roscd beginner_tutorials/
dhaval@dhaval-VirtualBox:~/catkin_ws/src/beginner_tutorials$ mkdir -p src
dhaval@dhaval-VirtualBox:~/catkin_ws/src/beginner_tutorials$ cd ~/catkin_ws/
dhaval@dhaval-VirtualBox:~/catkin_ws$ catkin_make
Base path: /home/dhaval/catkin_ws
Source space: /home/dhaval/catkin_ws/src
Build space: /home/dhaval/catkin_ws/build
Devel space: /home/dhaval/catkin_ws/devel
Install space: /home/dhaval/catkin_ws/install
WARNING: Package name "lab5Pkg" does not follow the naming conventions. It shoul
d start with a lower case letter and only contain lower case letters, digits, un
derscores, and dashes.
####
#### Running command: "make cmake_check_build_system" in "/home/dhaval/catkin_ws
/build"
####
-- Using CATKIN_DEVEL_PREFIX: /home/dhaval/catkin_ws/devel
-- Using CMAKE_PREFIX_PATH: /home/dhaval/catkin_ws/devel;/opt/ros/noetic
-- This workspace overlays: /home/dhaval/catkin_ws/devel;/opt/ros/noetic
-- Found PythonInterp: /usr/bin/python3 (found suitable version "3.8.10", minimu
m required is "3")
-- Using PYTHON_EXECUTABLE: /usr/bin/python3
-- Using Debian Python package layout
-- Using empy: /usr/lib/python3/dist-packages/em.py
```

- **Talker.cpp**

```
/*
 * Copyright (C) 2008, Morgan Quigley and Willow Garage, Inc.
 *
 * Redistribution and use in source and binary forms, with or without
 * modification, are permitted provided that the following conditions are
met:
 *   * Redistributions of source code must retain the above copyright
notice,
 *     this list of conditions and the following disclaimer.
 *   * Redistributions in binary form must reproduce the above copyright
 *     notice, this list of conditions and the following disclaimer in the
 *     documentation and/or other materials provided with the distribution.
 *   * Neither the names of Stanford University or Willow Garage, Inc. nor
the names of its
 *     contributors may be used to endorse or promote products derived from
 *     this software without specific prior written permission.
 *
 * THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
 * AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
 * IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
 * ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE
 * LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR
 * CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF
 * SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS
 * INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN
 * CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
 * ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE
 * POSSIBILITY OF SUCH DAMAGE.
 */
// %Tag(FULLTEXT)%
// %Tag(ROS_HEADER)%
#include "ros/ros.h"
// %EndTag(ROS_HEADER)%
// %Tag(MSG_HEADER)%
#include "std_msgs/String.h"
// %EndTag(MSG_HEADER)%
#include <sstream>
/**
 * This tutorial demonstrates simple sending of messages over the ROS system.
 */
int main(int argc, char **argv)
{
  /**
   * The ros::init() function needs to see argc and argv so that it can perform
   * any ROS arguments and name remapping that were provided at the command line.
   * For programmatic remappings you can use a different version of init() which takes
   * remappings directly, but for most command-line programs, passing argc and argv is
   * the easiest way to do it.  The third argument to init() is the name of the node.
   *
   * You must call one of the versions of ros::init() before using any other
   * part of the ROS system.
   */
// %Tag(INIT)%
  ros::init(argc, argv, "talker");
// %EndTag(INIT)%

  /**
   * NodeHandle is the main access point to communications with the ROS system.
   * The first NodeHandle constructed will fully initialize this node, and the last
   * NodeHandle destructed will close down the node.
   */
// %Tag(NODEHANDLE)%
  ros::NodeHandle n;
// %EndTag(NODEHANDLE)%
  /**
   * The advertise() function is how you tell ROS that you want to
   * publish on a given topic name. This invokes a call to the ROS
   * master node, which keeps a registry of who is publishing and who
   * is subscribing. After this advertise() call is made, the master
   * node will notify anyone who is trying to subscribe to this topic name,
   * and they will in turn negotiate a peer-to-peer connection with this
   * node.  advertise() returns a Publisher object which allows you to
   * publish messages on that topic through a call to publish().  Once
   * all copies of the returned Publisher object are destroyed, the topic
   * will be automatically unadvertised.
   *
   * The second parameter to advertise() is the size of the message queue
   * used for publishing messages.  If messages are published more quickly
   * than we can send them, the number here specifies how many messages to
   * buffer up before throwing some away.
   */
// %Tag(PUBLISHER)%
  ros::Publisher chatter_pub = n.advertise<std_msgs::String>("chatter",
1000);
// %EndTag(PUBLISHER)%
// %Tag(LOOP_RATE)%
  ros::Rate loop_rate(10);
// %EndTag(LOOP_RATE)%
  /**
   * A count of how many messages we have sent. This is used to create
   * a unique string for each message.
   */
// %Tag(ROS_OK)%
  int count = 0;
  while (ros::ok())
  {
// %EndTag(ROS_OK)%
    /**
     * This is a message object. You stuff it with data, and then publish
it.
     */
// %Tag(FILL_MESSAGE)%
    std_msgs::String msg;
    std::stringstream ss;
    ss << "hello world " << count;
    msg.data = ss.str();
// %EndTag(FILL_MESSAGE)%

// %Tag(ROSCONSOLE)%
    ROS_INFO("%s", msg.data.c_str());
// %EndTag(ROSCONSOLE)%
    /**
     * The publish() function is how you send messages. The parameter
     * is the message object. The type of this object must agree with the
type
     * given as a template parameter to the advertise<>() call, as was done
     * in the constructor above.
     */
// %Tag(PUBLISH)%
    chatter_pub.publish(msg);
// %EndTag(PUBLISH)%

// %Tag(SPINONCE)%
    ros::spinOnce();
// %EndTag(SPINONCE)%

// %Tag(RATE_SLEEP)%
    loop_rate.sleep();
// %EndTag(RATE_SLEEP)%
    ++count;
  }
  return 0;
}
// %EndTag(FULLTEXT)%
```

- **Listner.cpp**

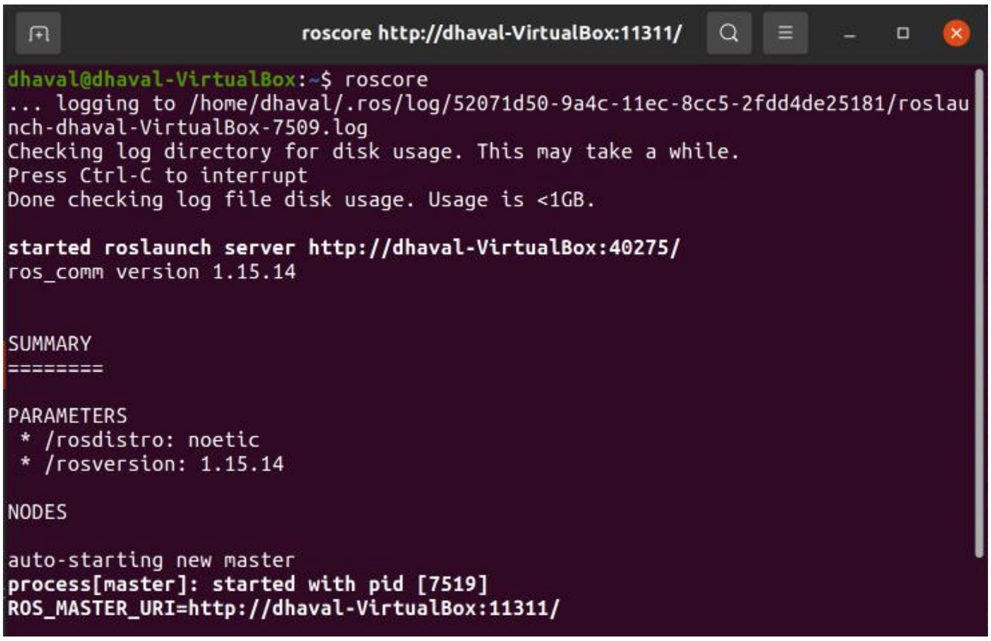
```
/*
 * Copyright (C) 2008, Morgan Quigley and Willow Garage, Inc.
 *
 * Redistribution and use in source and binary forms, with or without
 * modification, are permitted provided that the following conditions are
met:
 *   * Redistributions of source code must retain the above copyright
notice,
 *     this list of conditions and the following disclaimer.
 *   * Redistributions in binary form must reproduce the above copyright
 *     notice, this list of conditions and the following disclaimer in the
 *     documentation and/or other materials provided with the distribution.
 *   * Neither the names of Stanford University or Willow Garage, Inc. nor
the names of its
 *     contributors may be used to endorse or promote products derived from
 *     this software without specific prior written permission.
 *
 * THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS
IS"
 * AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO,
THE
 * IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR
PURPOSE
 * ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE
 * LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR
 * CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF
 * SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS
 * INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN
 * CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
 * ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF
THE
 * POSSIBILITY OF SUCH DAMAGE.
 */

// %Tag(FULLTEXT)%
```

```
#include "ros/ros.h"
#include "std_msgs/String.h"
/**
 * This tutorial demonstrates simple receipt of messages over the ROS
system.
 */
// %Tag(CALLBACK)%
void chatterCallback(const std_msgs::String::ConstPtr& msg)
{
  ROS_INFO("I heard: [%s]", msg->data.c_str());
}
// %EndTag(CALLBACK)%
int main(int argc, char **argv)
{
  /**
   * The ros::init() function needs to see argc and argv so that it can
perform
   * any ROS arguments and name remapping that were provided at the command
line.
   * For programmatic remappings you can use a different version of init()
which takes
   * remappings directly, but for most command-line programs, passing argc
and argv is
   * the easiest way to do it.  The third argument to init() is the name of
the node.
   *
   * You must call one of the versions of ros::init() before using any
other
   * part of the ROS system.
   */
  ros::init(argc, argv, "listener");
  /**
   * NodeHandle is the main access point to communications with the ROS
system.
   * The first NodeHandle constructed will fully initialize this node, and
the last
   * NodeHandle destructed will close down the node.
   */
  ros::NodeHandle n;
  /**
   * The subscribe() call is how you tell ROS that you want to receive
messages
   * on a given topic.  This invokes a call to the ROS
   * master node, which keeps a registry of who is publishing and who
   * is subscribing.  Messages are passed to a callback function, here
   * called chatterCallback.  subscribe() returns a Subscriber object that
you
   * must hold on to until you want to unsubscribe.  When all copies of the
Subscriber
   * object go out of scope, this callback will automatically be
unsubscribed from
   * this topic.
   *
   * The second parameter to the subscribe() function is the size of the
message
   * queue.  If messages are arriving faster than they are being processed,
this
```

```
   * is the number of messages that will be buffered up before beginning to
throw
   * away the oldest ones.
   */
// %Tag(SUBSCRIBER)%
  ros::Subscriber sub = n.subscribe("chatter", 1000, chatterCallback);
// %EndTag(SUBSCRIBER)%
  /**
   * ros::spin() will enter a loop, pumping callbacks.  With this version,
all
   * callbacks will be called from within this thread (the main one).
ros::spin()
   * will exit when Ctrl-C is pressed, or the node is shutdown by the
master.
   */
// %Tag(SPIN)%
  ros::spin();
// %EndTag(SPIN)%
  return 0;
}
// %EndTag(FULLTEXT)%
```

- **Now we will run** `$ roscore` **command in new terminal**

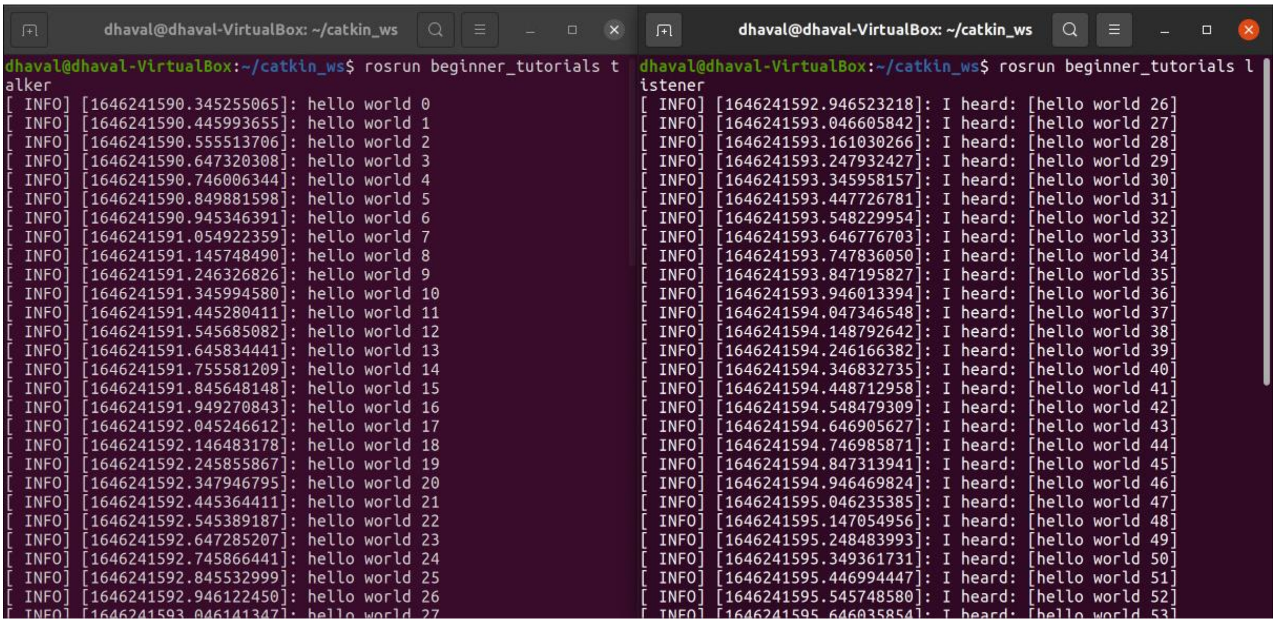
```
$ roscore
```



```
dhaval@dhaval-VirtualBox:~$ roscore
... logging to /home/dhaval/.ros/log/52071d50-9a4c-11ec-8cc5-2fdd4de25181/roslau
nch-dhaval-VirtualBox-7509.log
Checking log directory for disk usage. This may take a while.
Press Ctrl-C to interrupt
Done checking log file disk usage. Usage is <1GB.

started roslaunch server http://dhaval-VirtualBox:40275/
ros_comm version 1.15.14


SUMMARY
========

PARAMETERS
 * /rosdistro: noetic
 * /rosversion: 1.15.14

NODES

auto-starting new master
process[master]: started with pid [7519]
ROS_MASTER_URI=http://dhaval-VirtualBox:11311/
```

- **Now we will run publisher and subscriber in different terminal windows**

```
$ rosrun beginner_tutorials talker
$ rosrun beginner_tutorials listener
```



dhaval@dhaval-VirtualBox: ~/catkin_ws
dhaval@dhaval-VirtualBox: ~/catkin_ws

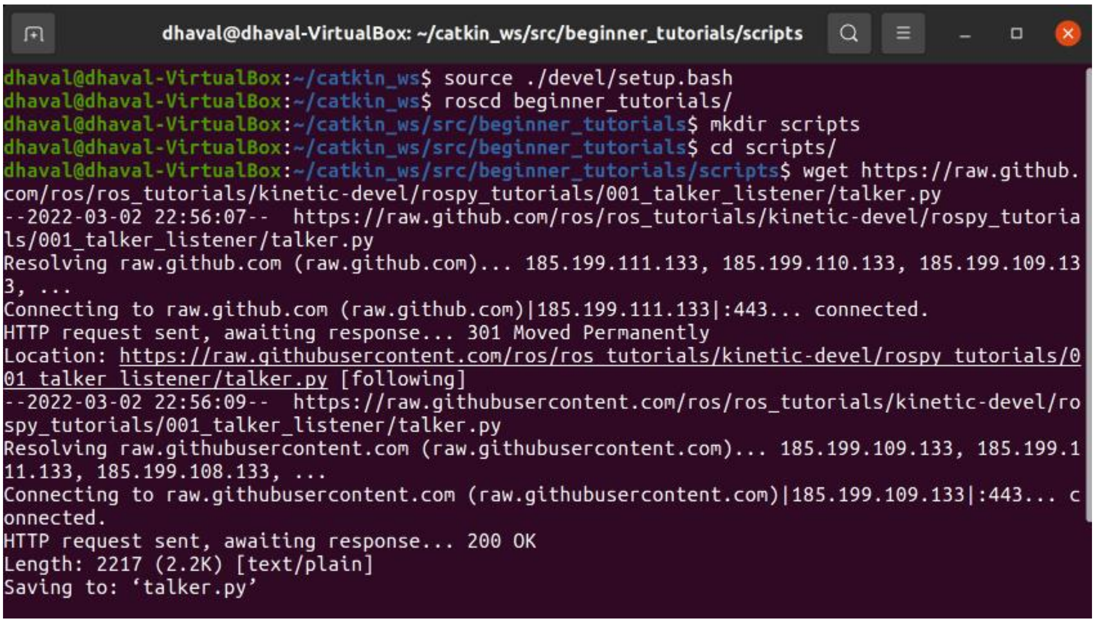
```
dhaval@dhaval-VirtualBox:~/catkin_ws$ rosrun beginner_tutorials t
alker
[ INFO] [1646241590.345255065]: hello world 0
[ INFO] [1646241590.445993655]: hello world 1
[ INFO] [1646241590.555513706]: hello world 2
[ INFO] [1646241590.647320308]: hello world 3
[ INFO] [1646241590.746006344]: hello world 4
[ INFO] [1646241590.849881598]: hello world 5
[ INFO] [1646241590.945346391]: hello world 6
[ INFO] [1646241591.054922359]: hello world 7
[ INFO] [1646241591.145748490]: hello world 8
[ INFO] [1646241591.246326826]: hello world 9
[ INFO] [1646241591.345994580]: hello world 10
[ INFO] [1646241591.445280411]: hello world 11
[ INFO] [1646241591.545685082]: hello world 12
[ INFO] [1646241591.645834441]: hello world 13
[ INFO] [1646241591.755581209]: hello world 14
[ INFO] [1646241591.845648148]: hello world 15
[ INFO] [1646241591.949270843]: hello world 16
[ INFO] [1646241592.045246612]: hello world 17
[ INFO] [1646241592.146483178]: hello world 18
[ INFO] [1646241592.245855867]: hello world 19
[ INFO] [1646241592.347946795]: hello world 20
[ INFO] [1646241592.445364411]: hello world 21
[ INFO] [1646241592.545389187]: hello world 22
[ INFO] [1646241592.647285207]: hello world 23
[ INFO] [1646241592.745866441]: hello world 24
[ INFO] [1646241592.845532999]: hello world 25
[ INFO] [1646241592.946122450]: hello world 26
[ INFO] [1646241593.046141347]: hello world 27
```

```
dhaval@dhaval-VirtualBox:~/catkin_ws$ rosrun beginner_tutorials l
istener
[ INFO] [1646241592.946523218]: I heard: [hello world 26]
[ INFO] [1646241593.046605842]: I heard: [hello world 27]
[ INFO] [1646241593.161030266]: I heard: [hello world 28]
[ INFO] [1646241593.247932427]: I heard: [hello world 29]
[ INFO] [1646241593.345958157]: I heard: [hello world 30]
[ INFO] [1646241593.447726781]: I heard: [hello world 31]
[ INFO] [1646241593.548229954]: I heard: [hello world 32]
[ INFO] [1646241593.646776703]: I heard: [hello world 33]
[ INFO] [1646241593.747836050]: I heard: [hello world 34]
[ INFO] [1646241593.847195827]: I heard: [hello world 35]
[ INFO] [1646241593.946013394]: I heard: [hello world 36]
[ INFO] [1646241594.047346548]: I heard: [hello world 37]
[ INFO] [1646241594.148792642]: I heard: [hello world 38]
[ INFO] [1646241594.246166382]: I heard: [hello world 39]
[ INFO] [1646241594.346832735]: I heard: [hello world 40]
[ INFO] [1646241594.448712958]: I heard: [hello world 41]
[ INFO] [1646241594.548479309]: I heard: [hello world 42]
[ INFO] [1646241594.646905627]: I heard: [hello world 43]
[ INFO] [1646241594.746985871]: I heard: [hello world 44]
[ INFO] [1646241594.847313941]: I heard: [hello world 45]
[ INFO] [1646241594.946469824]: I heard: [hello world 46]
[ INFO] [1646241595.046235385]: I heard: [hello world 47]
[ INFO] [1646241595.147054956]: I heard: [hello world 48]
[ INFO] [1646241595.248483993]: I heard: [hello world 49]
[ INFO] [1646241595.349361731]: I heard: [hello world 50]
[ INFO] [1646241595.446994447]: I heard: [hello world 51]
[ INFO] [1646241595.545748580]: I heard: [hello world 52]
[ INFO] [1646241595.646035854]: I heard: [hello world 53]
```

PYTHON

- **First lets create a 'scripts' folder to store our Python scripts in:**

```
$ mkdir scripts
$ cd scripts
```



dhaval@dhaval-VirtualBox: ~/catkin_ws/src/beginner_tutorials/scripts

```
dhaval@dhaval-VirtualBox:~/catkin_ws$ source ./devel/setup.bash
dhaval@dhaval-VirtualBox:~/catkin_ws$ roscd beginner_tutorials/
dhaval@dhaval-VirtualBox:~/catkin_ws/src/beginner_tutorials$ mkdir scripts
dhaval@dhaval-VirtualBox:~/catkin_ws/src/beginner_tutorials$ cd scripts/
dhaval@dhaval-VirtualBox:~/catkin_ws/src/beginner_tutorials/scripts$ wget https://raw.github.
com/ros/ros_tutorials/kinetic-devel/rospy_tutorials/001_talker_listener/talker.py
--2022-03-02 22:56:07--  https://raw.github.com/ros/ros_tutorials/kinetic-devel/rospy_tutoria
ls/001_talker_listener/talker.py
Resolving raw.github.com (raw.github.com)... 185.199.111.133, 185.199.110.133, 185.199.109.13
3, ...
Connecting to raw.github.com (raw.github.com)|185.199.111.133|:443... connected.
HTTP request sent, awaiting response... 301 Moved Permanently
Location: https://raw.githubusercontent.com/ros/ros_tutorials/kinetic-devel/rospy_tutorials/0
01_talker_listener/talker.py [following]
--2022-03-02 22:56:09--  https://raw.githubusercontent.com/ros/ros_tutorials/kinetic-devel/ro
spy_tutorials/001_talker_listener/talker.py
Resolving raw.githubusercontent.com (raw.githubusercontent.com)... 185.199.109.133, 185.199.1
11.133, 185.199.108.133, ...
Connecting to raw.githubusercontent.com (raw.githubusercontent.com)|185.199.109.133|:443... c
onnected.
HTTP request sent, awaiting response... 200 OK
Length: 2217 (2.2K) [text/plain]
Saving to: ‘talker.py’
```

- **Talker.py**

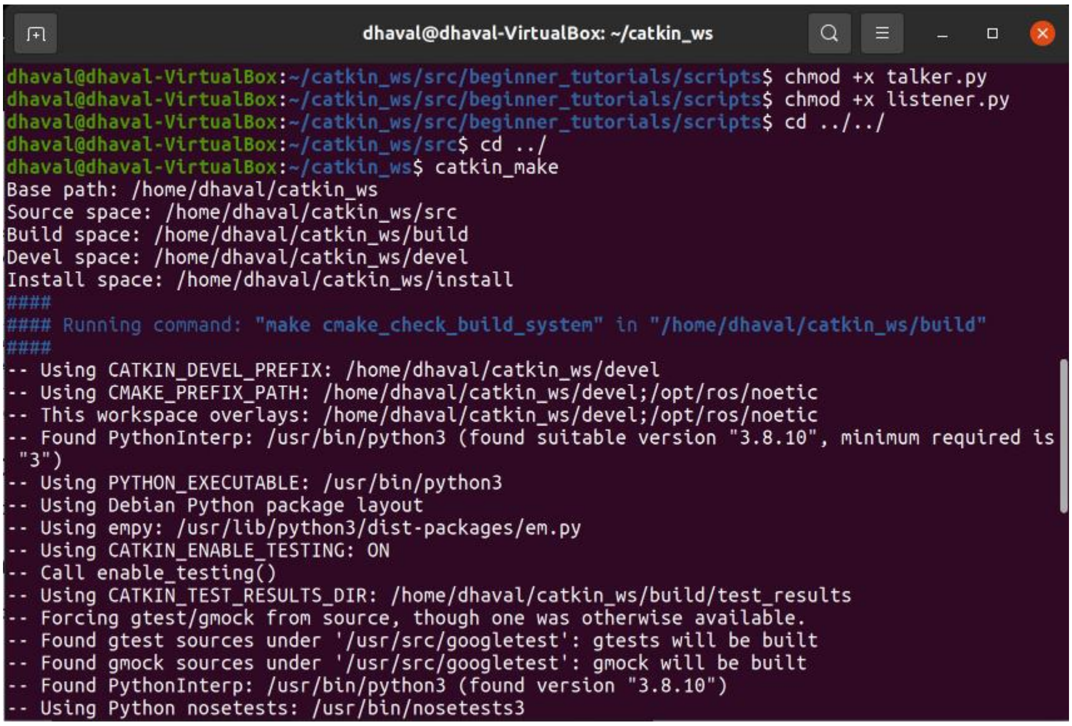
```
#!/usr/bin/env python
# Software License Agreement (BSD License)
#
# Copyright (c) 2008, Willow Garage, Inc.
# All rights reserved.
#
# Redistribution and use in source and binary forms, with or without
# modification, are permitted provided that the following conditions
# are met:
#
#  * Redistributions of source code must retain the above copyright
#    notice, this list of conditions and the following disclaimer.
#  * Redistributions in binary form must reproduce the above
#    copyright notice, this list of conditions and the following
#    disclaimer in the documentation and/or other materials provided
#    with the distribution.
#  * Neither the name of Willow Garage, Inc. nor the names of its
#    contributors may be used to endorse or promote products derived
#    from this software without specific prior written permission.
#
# THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
# "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
# LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS
# FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE
# COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT,
# INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING,
# BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES;
# LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER
# CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
# LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN
# ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE
# POSSIBILITY OF SUCH DAMAGE.
#
# Revision $Id$
## Simple talker demo that published std_msgs/Strings messages
## to the 'chatter' topic
import rospy
from std_msgs.msg import String
def talker():
    pub = rospy.Publisher('chatter', String, queue_size=10)
    rospy.init_node('talker', anonymous=True)
    rate = rospy.Rate(10) # 10hz
    while not rospy.is_shutdown():
        hello_str = "hello world %s" % rospy.get_time()
        rospy.loginfo(hello_str)
        pub.publish(hello_str)
        rate.sleep()
if __name__ == '__main__':
    try:
        talker()
    except rospy.ROSInterruptException:
        pass
```

- **Lister.py**

```
#!/usr/bin/env python
# Software License Agreement (BSD License)
#
# Copyright (c) 2008, Willow Garage, Inc.
# All rights reserved.
#
# Redistribution and use in source and binary forms, with or without
# modification, are permitted provided that the following conditions
# are met:
#
#  * Redistributions of source code must retain the above copyright
#    notice, this list of conditions and the following disclaimer.
#  * Redistributions in binary form must reproduce the above
#    copyright notice, this list of conditions and the following
#    disclaimer in the documentation and/or other materials provided
#    with the distribution.
#  * Neither the name of Willow Garage, Inc. nor the names of its
#    contributors may be used to endorse or promote products derived
#    from this software without specific prior written permission.
#
# THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
# "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
# LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS
# FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE
# COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT,
# INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING,
# BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES;
# LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER
# CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
# LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN
# ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE
# POSSIBILITY OF SUCH DAMAGE.
#
# Revision $Id$
## Simple talker demo that listens to std_msgs/Strings published
## to the 'chatter' topic
import rospy
from std_msgs.msg import String
def callback(data):
    rospy.loginfo(rospy.get_caller_id() + 'I heard %s', data.data)
def listener():
    # In ROS, nodes are uniquely named. If two nodes with the same
    # name are launched, the previous one is kicked off. The
    # anonymous=True flag means that rospy will choose a unique
    # name for our 'listener' node so that multiple listeners can
    # run simultaneously.
    rospy.init_node('listener', anonymous=True)
    rospy.Subscriber('chatter', String, callback)
    # spin() simply keeps python from exiting until this node is stopped
    rospy.spin()
if __name__ == '__main__':
    listener()
```

- **Now making talker.py and listener.py executable:**

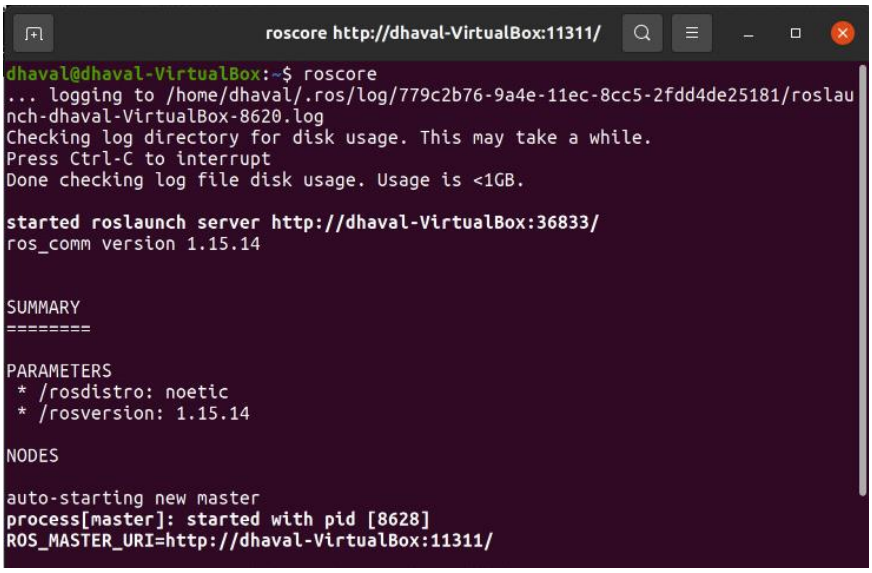
```
$ chmod +x talker.py

$ chmod +x listener.py
```



```
dhaval@dhaval-VirtualBox:~/catkin_ws/src/beginner_tutorials/scripts$ chmod +x talker.py
dhaval@dhaval-VirtualBox:~/catkin_ws/src/beginner_tutorials/scripts$ chmod +x listener.py
dhaval@dhaval-VirtualBox:~/catkin_ws/src/beginner_tutorials/scripts$ cd ../../
dhaval@dhaval-VirtualBox:~/catkin_ws/src$ cd ../
dhaval@dhaval-VirtualBox:~/catkin_ws$ catkin_make
Base path: /home/dhaval/catkin_ws
Source space: /home/dhaval/catkin_ws/src
Build space: /home/dhaval/catkin_ws/build
Devel space: /home/dhaval/catkin_ws/devel
Install space: /home/dhaval/catkin_ws/install
####
#### Running command: "make cmake_check_build_system" in "/home/dhaval/catkin_ws/build"
####
-- Using CATKIN_DEVEL_PREFIX: /home/dhaval/catkin_ws/devel
-- Using CMAKE_PREFIX_PATH: /home/dhaval/catkin_ws/devel;/opt/ros/noetic
-- This workspace overlays: /home/dhaval/catkin_ws/devel;/opt/ros/noetic
-- Found PythonInterp: /usr/bin/python3 (found suitable version "3.8.10", minimum required is
 "3")
-- Using PYTHON_EXECUTABLE: /usr/bin/python3
-- Using Debian Python package layout
-- Using empy: /usr/lib/python3/dist-packages/em.py
-- Using CATKIN_ENABLE_TESTING: ON
-- Call enable_testing()
-- Using CATKIN_TEST_RESULTS_DIR: /home/dhaval/catkin_ws/build/test_results
-- Forcing gtest/gmock from source, though one was otherwise available.
-- Found gtest sources under '/usr/src/googletest': gtests will be built
-- Found gmock sources under '/usr/src/googletest': gmock will be built
-- Found PythonInterp: /usr/bin/python3 (found version "3.8.10")
-- Using Python nosetests: /usr/bin/nosetests3
```

- **Now we will run** `$ roscore` **command in new terminal**

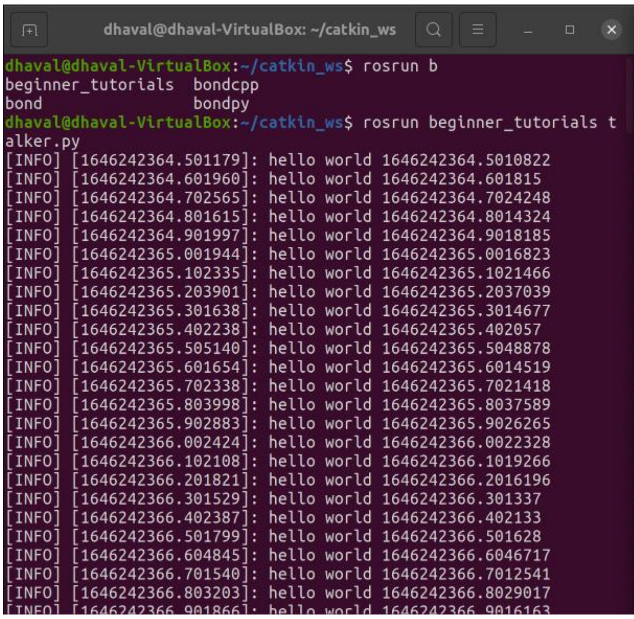
```
$ roscore
```



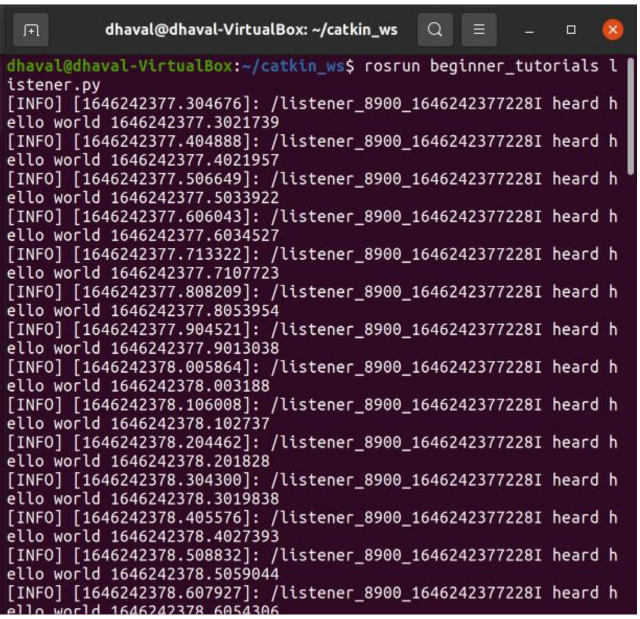
```
dhaval@dhaval-VirtualBox:~$ roscore
... logging to /home/dhaval/.ros/log/779c2b76-9a4e-11ec-8cc5-2fdd4de25181/roslau
nch-dhaval-VirtualBox-8620.log
Checking log directory for disk usage. This may take a while.
Press Ctrl-C to interrupt
Done checking log file disk usage. Usage is <1GB.

started roslaunch server http://dhaval-VirtualBox:36833/
ros_comm version 1.15.14


SUMMARY
========

PARAMETERS
 * /rosdistro: noetic
 * /rosversion: 1.15.14

NODES

auto-starting new master
process[master]: started with pid [8628]
ROS_MASTER_URI=http://dhaval-VirtualBox:11311/
```

- **Now we will run publisher and subscriber in different terminal windows**

```
$ rosrun beginner_tutorials talker.py

$ rosrun beginner_tutorials listener.py
```



```
dhaval@dhaval-VirtualBox:~/catkin_ws$ rosrun b
beginner_tutorials  bondcpp
bond                bondpy
dhaval@dhaval-VirtualBox:~/catkin_ws$ rosrun beginner_tutorials t
alker.py
[INFO] [1646242364.501179]: hello world 1646242364.5010822
[INFO] [1646242364.601960]: hello world 1646242364.601815
[INFO] [1646242364.702565]: hello world 1646242364.7024248
[INFO] [1646242364.801615]: hello world 1646242364.8014324
[INFO] [1646242364.901997]: hello world 1646242364.9018185
[INFO] [1646242365.001944]: hello world 1646242365.0016823
[INFO] [1646242365.102335]: hello world 1646242365.1021466
[INFO] [1646242365.203901]: hello world 1646242365.2037039
[INFO] [1646242365.301638]: hello world 1646242365.3014677
[INFO] [1646242365.402238]: hello world 1646242365.402057
[INFO] [1646242365.505140]: hello world 1646242365.5048878
[INFO] [1646242365.601654]: hello world 1646242365.6014519
[INFO] [1646242365.702338]: hello world 1646242365.7021418
[INFO] [1646242365.803998]: hello world 1646242365.8037589
[INFO] [1646242365.902883]: hello world 1646242365.9026265
[INFO] [1646242366.002424]: hello world 1646242366.0022328
[INFO] [1646242366.102108]: hello world 1646242366.1019266
[INFO] [1646242366.201821]: hello world 1646242366.2016196
[INFO] [1646242366.301529]: hello world 1646242366.301337
[INFO] [1646242366.402387]: hello world 1646242366.402133
[INFO] [1646242366.501799]: hello world 1646242366.501628
[INFO] [1646242366.604845]: hello world 1646242366.6046717
[INFO] [1646242366.701540]: hello world 1646242366.7012541
[INFO] [1646242366.803203]: hello world 1646242366.8029017
[INFO] [1646242366.901866]: hello world 1646242366.9016163
```



```
dhaval@dhaval-VirtualBox:~/catkin_ws$ rosrun beginner_tutorials l
istener.py
[INFO] [1646242377.304676]: /listener_8900_1646242377228I heard h
ello world 1646242377.3021739
[INFO] [1646242377.404888]: /listener_8900_1646242377228I heard h
ello world 1646242377.4021957
[INFO] [1646242377.506649]: /listener_8900_1646242377228I heard h
ello world 1646242377.5033922
[INFO] [1646242377.606043]: /listener_8900_1646242377228I heard h
ello world 1646242377.6034527
[INFO] [1646242377.713322]: /listener_8900_1646242377228I heard h
ello world 1646242377.7107723
[INFO] [1646242377.808209]: /listener_8900_1646242377228I heard h
ello world 1646242377.8053954
[INFO] [1646242377.904521]: /listener_8900_1646242377228I heard h
ello world 1646242377.9013038
[INFO] [1646242378.005864]: /listener_8900_1646242377228I heard h
ello world 1646242378.003188
[INFO] [1646242378.106008]: /listener_8900_1646242377228I heard h
ello world 1646242378.102737
[INFO] [1646242378.204462]: /listener_8900_1646242377228I heard h
ello world 1646242378.201828
[INFO] [1646242378.304300]: /listener_8900_1646242377228I heard h
ello world 1646242378.3019838
[INFO] [1646242378.405576]: /listener_8900_1646242377228I heard h
ello world 1646242378.4027393
[INFO] [1646242378.508832]: /listener_8900_1646242377228I heard h
ello world 1646242378.5059044
[INFO] [1646242378.607927]: /listener_8900_1646242377228I heard h
ello world 1646242378.6054306
```